

Claim Listing

1. (Currently Amended) A method of [[improved]] resource arbitration, the method comprising the steps of:
 - defining four priority classes, managed high (MH), managed low (ML), opportunistic high (OH), and opportunistic low (OL), to access one or more resources;
 - assigning a priority class to each resource access request;
 - creating an access request concentrator (ARC) for each resource, through which each resource is accessed;
 - choosing a resource access request at each ARC using the priority order MH, ML, OH, and OL, in decreasing order of priority;
 - determining whether OH priority class resource access requests are locked out;
 - upon a determination that OH priority class resource access requests are locked out, temporarily changing the priority order to OH, OL, MH, and ML, in decreasing order of priority;
 - determining whether OL priority class resource access requests are locked out; and
 - upon a determination that OL priority class resource access requests are locked out, temporarily changing the priority order to MH, OL, OH, and ML, in decreasing order of priority.
2. (Original) The method of Claim 1, wherein the access request concentrator comprises at least one arbitration point.
3. (Original) The method of Claim 2, wherein the resource being arbitrated is bandwidth.

4. (Original) The method of Claim 3, wherein the resource access request is a command.
5. (Original) The method of Claim 3, wherein the resource access request is a data component associated with a command.
6. (Currently Amended) A method of arbitrating between a plurality of resource access requests, ~~using at least one arbitration point~~, the method comprising the steps of:
 - defining four priority classes, managed high (MH), managed low (ML), opportunistic high (OH), and opportunistic low (OL);
 - assigning a priority class to each of the resource access requests;
 - dividing the resource access requests into subgroups;
 - arbitrating between the resource access requests in each subgroup ~~using an arbitration point~~ to obtain a smaller set of the resource access requests, wherein the arbitrating comprises:
 - choosing one of the resource access requests using the priority order MH, ML, OH, and OL, in decreasing order of priority;
 - determining whether OH priority class resource access requests are locked out;
 - upon a determination that OH priority class resource access requests are locked out,
 - temporarily changing the priority order to OH, OL, MH, and ML, in decreasing order of priority;
 - determining whether OL priority class resource access requests are locked out;

upon a determination that OL priority class resource access requests are locked out,
temporarily changing the priority order to MH, OL, OH, and ML, in
decreasing order of priority; and
~~repeating the above two steps for the smaller set of resource access requests until a single~~
~~resource access request remains~~
arbitrating between the resource access requests in the smaller set to obtain a single one of
the resource access requests.

7. (Currently Amended) An apparatus for providing [[improved]] resource arbitration, the apparatus comprising:

means for defining four priority classes, managed high (MH), managed low (ML),
opportunistic high (OH), and opportunistic low (OL), to access one or more
resources;

means for assigning a priority class to each resource access request;

means for creating an access request concentrator (ARC) for each resource, through which
each resource is accessed;

means for choosing an access request at each ARC using the priority order MH, ML, OH,
and OL, in decreasing order of priority;

means for determining whether OH priority class resource access requests are locked out;

upon a determination that OH priority class resource access requests are locked out, means
for temporarily changing the priority order to OH, OL, MH, and ML, in decreasing
order of priority;

means for determining whether OL priority class resource access requests are locked out;

and

upon a determination that OL priority class resource access requests are locked out, means

for temporarily changing the priority order to MH, OL, OH, and ML, in decreasing

order of priority.

8. (Original) The apparatus of Claim 7 wherein the access request concentrator comprises at least one arbitration point.

9. (Original) The apparatus of Claim 8 wherein the resource being arbitrated is bandwidth.

10. (Original) The apparatus of Claim 9 wherein the resource access request is a command.

11. (Original) The apparatus of Claim 9 wherein the resource access request is a data component associated with a command.

12. (Currently Amended) An apparatus for arbitrating between a plurality of resource access requests, ~~using at least one arbitration point, the apparatus~~ comprising:

means for defining four priority classes, managed high (MH), managed low (ML),

opportunistic high (OH), and opportunistic low (OL);

means for assigning a priority class to each of the resource access requests;

means for dividing the resource access requests into subgroups;

means for arbitrating between the resource access requests in each subgroup ~~using an arbitration point~~ to obtain a smaller set of the resource access requests, wherein the arbitrating comprises:

choosing one of the resource access requests using the priority order MH, ML, OH, and OL, in decreasing order of priority;

determining whether OH priority class resource access requests are locked out;

upon a determination that OH priority class resource access requests are locked out,

temporarily changing the priority order to OH, OL, MH, and ML, in decreasing order of priority;

determining whether OL priority class resource access requests are locked out;

upon a determination that OL priority class resource access requests are locked out,

temporarily changing the priority order to MH, OL, OH, and ML, in decreasing order of priority; and

means for ~~repeating the above two steps for the smaller set of resource access requests until a single resource access request remains~~ arbitrating between the resource access requests in the smaller set to obtain a single one of the resource access requests.

13. (Currently Amended) A computer program product for providing [[improved]] resource arbitration, the computer program product having a medium with a computer program embodied thereon, the computer program comprising:

computer program code for defining four priority classes, managed high (MH), managed low (ML), opportunistic high (OH), and opportunistic low (OL), to access one or more resources;

computer program code for assigning a priority class to each resource access request;
computer program code for creating an access request concentrator (ARC) for each
resource, through which each resource is accessed;
computer program code for choosing an access request at each ARC using the priority order
MH, ML, OH, and OL, in decreasing order of priority;
computer program code for determining whether OH priority class resource access requests
are locked out;
upon a determination that OH priority class resource access requests are locked out,
computer program code for temporarily changing the priority order to OH, OL, MH,
and ML, in decreasing order of priority;
computer program code for determining whether OL priority class resource access requests
are locked out; and
upon a determination that OL priority class resource access requests are locked out,
computer program code for temporarily changing the priority order to MH, OL, OH,
and ML, in decreasing order of priority.

14. (Original) The computer program product of Claim 13 wherein the access request concentrator comprises at least one arbitration point.

15. (Original) The computer program product of Claim 14 wherein the resource being arbitrated is bandwidth.

16. (Original) The computer program product of Claim 15 wherein the resource access request is a command.

17. (Original) The computer program product of Claim 15 wherein the resource access request is a data component associated with a command.

18. (Currently Amended) A computer program product for arbitrating between a plurality of resource access requests, ~~using at least one arbitration point~~, the computer program product having a medium with a computer program embodied thereon, the computer program comprising:

computer program code for defining four priority classes, managed high (MH), managed

low (ML), opportunistic high (OH), and opportunistic low (OL);

computer program code for assigning a priority class to each of the resource access requests;

computer program code for dividing the resource access requests into subgroups;

computer program code for arbitrating between the resource access requests in each

subgroup ~~using an arbitration point~~ to obtain a smaller set of the resource access requests, wherein the arbitrating comprises:

choosing one of the resource access requests using the priority order MH, ML, OH,

and OL, in decreasing order of priority;

determining whether OH priority class resource access requests are locked out;

upon a determination that OH priority class resource access requests are locked out,

temporarily changing the priority order to OH, OL, MH, and ML, in

decreasing order of priority;

determining whether OL priority class resource access requests are locked out;

upon a determination that OL priority class resource access requests are locked out,
temporarily changing the priority order to MH, OL, OH, and ML, in decreasing
order of priority; and
computer program code for ~~repeating the above two steps for the smaller set of resource~~
~~access requests until a single resource access request remains~~ arbitrating between the
resource access requests in the smaller set to obtain a single one of the resource
access requests.